

MA501G Diskret matematik och logik, HT 2025

MATLABUPPGIFTER 4 (AV 4)

Lösningarna ska redovisas vid datorterminal, i par (rekommenderas) eller enskilt, vid någon av de schemalagda datorövningarna. För att bli godkänd ska du visa förståelse för och kunna resonera kring såväl er lösning som de relaterade teoretiska begreppen. Det är tillåtet att diskutera uppgifterna med era kurskamrater, men inte att kopiera kod från varandra eller från andra källor.

I dessa uppgifter kommer ni att arbeta med ändliga automater i Matlab. *Ta för vana att alltid testköra era funktioner.* Titta i *Introduktion till MATLAB* om ni behöver repetera något. Jag påminner också om Matlabs funktion `help`; om ni skriver t.ex. `help ==` i kommandofönstret så berättar Matlab vad funktionen `==` gör.

1. Ändliga automater kan enkelt implementeras i Matlab. Vi börjar med accepterande automater. För att definiera själva automaten tänker oss att tillstånden heter q_1, q_2, \dots, q_n , där vi i Matlab låter talet j beteckna tillståndet q_j . Vårt alfabet är $1, 2, \dots, k$. För att definiera en automat behöver vi ange en övergångsmatris av storlek $n \times k$, ett starttillstånd och en lista över accepterande tillstånd, som skrivs som en vektor av längd n där vi skriver 1 på position j om q_j är accepterande och 0 annars. En automat som känner igen strängar i alfabetet $\{1, 2\}$ som slutar med två ettor kan då skrivas med övergångsmatrisen

$$\delta = \begin{pmatrix} 2 & 1 \\ 3 & 1 \\ 3 & 1 \end{pmatrix},$$

de accepterande tillstånden $F = (0, 0, 1)$ (vilket innebär att q_3 är det enda accepterande tillståndet) och starttillstånd 1 (vilket alltså tolkas som q_1).

2. Vi ska nu skapa en *rekursiv* funktion som utifrån övergångsmatrisen, de accepterande tillstånden, starttillstånd och en textsträng (i vårt fall skriven som en vektor) avgör om strängen ska accepteras. Tänk igenom hur starttillstånd och textsträng ändras när vi läser första tecknet i strängen så blir rekursionssteget tydligt. Implementera denna rekursiva funktion och testa den med ovanstående exempel.
3. Betrakta det reguljära språket

$$A = \{1, 121, 12121, 1212121, \dots\} \cup \{2, 212, 21212, 2121212, \dots\}.$$

Ange ett reguljärt uttryck som beskriver A och rita en accepterande automat som känner igen A . Implementera sedan denna i Matlab (det vill säga bestäm och skriv in övergångsmatris, lista över accepterande tillstånd och starttillstånd) och testa med din rekursiva funktion att automaten verkligen känner igen A .

4. Skapa nu en rekursiv funktion som kan hantera utmatande automater. Vi ersätter då vektorn med accepterande tillstånd med en $n \times k$ -matris som anger vilken utmatning som respektive övergång genererar.
5. Skapa en utmatande automat som tar in strängar av ettor, tvåor och treor och skickar ut en etta varje gång summan av de inlästa talen är delbar med 3, och en tvåa annars. Till exempel ska indata 1312311321 ge utdata 222221121.